Attribute-Enhanced Face Recognition with Neural Tensor Fusion Networks

Guosheng Hu[1] Yang Hua[1,2] Yang Yuan[1] Zhihong Zhang[3] Zheng Lu[1]
Sankha S. Mukherjee[1] Timothy M. Hospedales[4] Neil M. Robertson[1,2] Yongxin Yang[5,6]
[1]AnyVision [2]Queen's University Belfast [3]Xiamen University
[4]The University of Edinburgh [5]Queen Mary University of London [6]Yang's Accounting Consultancy Ltd
{guosheng.hu,yang.hua,yuany,steven,rick}@anyvision.co.uk, N.Robertson@qub.ac.uk
zhihong@xmu.edu.cn, t.hospedales@ed.ac.uk, yongxin@yang.ac

Abstract

Deep learning has achieved great success in face recognition, however deep-learned features still have limited invariance to strong intra-personal variations such as large pose changes. It is observed that some facial attributes (e.g. eyebrow thickness, gender) are robust to such variations. We present the first work to systematically *explore how the fusion of face recognition features (FRF) and facial attribute features (FAF) can enhance face recognition performance in various challenging scenarios. Despite the promise of FAF, we find that in practice existing fusion methods fail to leverage FAF to boost face recognition performance in some challenging scenarios. Thus, we develop a powerful tensor-based framework which formulates feature fusion as a tensor optimisation problem. It is non-trivial to directly optimise this tensor due to the large number of parameters to optimise. To solve this problem, we establish a theoretical equivalence between low-rank tensor optimisation and a two-stream gated neural network. This equivalence allows tractable learning using standard neural network optimisation tools, leading to accurate and stable optimisation. Experimental results show the fused feature works better than individual features, thus proving for the first time that facial attributes aid face recognition. We achieve state-of-the-art performance on three popular databases: MultiPIE (cross pose, lighting and expression), CASIA NIR-VIS2.0 (cross-modality environment) and LFW (uncontrolled environment).*

1. Introduction

Face recognition has advanced dramatically with the advent of bigger datasets, and improved methodologies for generating features that are variant to identity but invariant to covariates such as pose, expression and illumination. Deep learning methodologies [41, 40, 42, 32] have proven particularly effective recently, thanks to end-to-end representation learning with a discriminative face recognition objective. Nevertheless, the resulting features still show imperfect invariance to the strong intra-personal variations in real-world scenarios. We observe that facial attributes provide a robust invariant cue in such challenging scenarios. For example gender and ethnicity are likely to be invariant to pose and expression, while eyebrow thickness may be invariant to lighting and resolution. Overall, face recognition features (FRF) are very discriminative but less robust; while facial attribute features (FAF) are robust but less discriminative. Thus these two features are potentially complementary, *if* a suitable fusion method can be devised. To the best of our knowledge, we are the first to systematically explore the fusion of FAF and FRF in various face recognition scenarios. We empirically show that this fusion can greatly enhance face recognition performance.

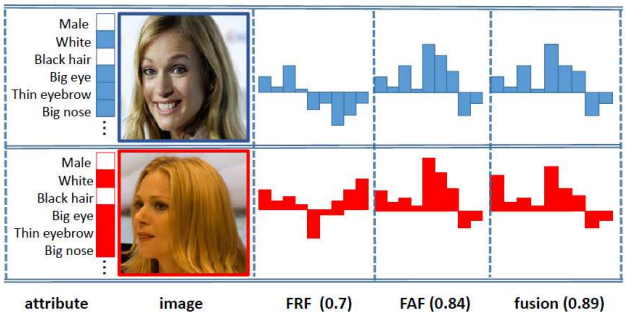

Figure 1: A sample attribute list is given (*col.1*) which pertains to the images of the same individual at different poses (*col.2*). While the similarity scores for each dimension vary in the face recognition feature (FRF) set (*col.3*), the face attribute feature (FAF) set (*col.4*) remains very similar. The fused features (*col.5*) are more similar and a higher similarity score (0.89) is achieved.

Though facial attributes are an important cue for face recognition, in practice, we find the existing fusion methods including early (feature) or late (score) fusion cannot reliably improve the performance [34]. In particular, while

offering some robustness, FAF is generally less discriminative than FRF. Existing methods cannot synergistically fuse such asymmetric features, and usually lead to worse performance than achieved by the stronger feature (FRF) only. In this work, we propose a novel tensor-based fusion framework that is uniquely capable of fusing the very asymmetric FAF and FRF. Our framework provides a more powerful and robust fusion approach than existing strategies by learning from all interactions between the two feature views. To train the tensor in a tractable way given the large number of required parameters, we formulate the optimisation with an identity-supervised objective by constraining the tensor to have a low-rank form. We establish an equivalence between this low-rank tensor and a two-stream gated neural network. Given this equivalence, the proposed tensor is easily optimised with standard deep neural network toolboxes. Our technical contributions are:

- It is the first work to *systematically* investigate and verify that facial attributes are an important cue in various face recognition scenarios. In particular, we investigate face recognition with extreme pose variations, i.e. $\pm 90^{\circ}$ from frontal, showing that attributes are important for performance enhancement.
- A rich tensor-based fusion framework is proposed. We show the low-rank Tucker-decomposition of this tensor-based fusion has an equivalent Gated Two-stream Neural Network (GTNN), allowing easy yet effective optimisation by neural network learning. In addition, we bring insights from neural networks into the field of tensor optimisation. The code is available: https://github.com/yanghuadr/Neural-Tensor-Fusion-Network
- We achieve state-of-the-art face recognition performance using the fusion of face (newly designed 'LeanFace' deep learning feature) and attribute-based features on three popular databases: MultiPIE (controlled environment), CASIA NIR-VIS2.0 (cross-modality environment) and LFW (uncontrolled environment).

2. Related Work

Face Recognition. The face representation (feature) is the most important component in contemporary face recognition system. There are two types: hand-crafted and deep learning features.

Widely used hand-crafted face descriptors include Local Binary Pattern (LBP) [26], Gabor filters [23], etc. Compared to pixel values, these features are variant to identity and relatively invariant to intra-personal variations, and thus they achieve promising performance in controlled environments. However, they perform less well on face recognition in uncontrolled environments (FRUE). There are two main routes to improve FRUE performance with hand-crafted features, one is to use very high dimensional features (dense sampling features) [5] and the other is to enhance the features with downstream metric learning.

Unlike hand-crafted features where (in)variances are engineered, deep learning features learn the (in)variances from data. Recently, convolutional neural networks (CNNs) achieved impressive results on FRUE. DeepFace [44], a carefully designed 8-layer CNN, is an early landmark method. Another well-known line of work is DeepID [41] and its variants DeepID2 [40], DeepID2+ [42]. The DeepID family uses an ensemble of many small CNNs trained independently using different facial patches to improve the performance. In addition, some CNNs originally designed for object recognition, such as VGGNet [38] and Inception [43], were also used for face recognition [29, 32]. Most recently, a center loss [47] is introduced to learn more discriminative features.

Facial Attribute Recognition. Facial attribute recognition (FAR) is also well studied. A notable early study [21] extracted carefully designed hand-crafted features including aggregations of colour spaces and image gradients, before training an independent SVM to detect each attribute. As for face recognition, deep learning features now outperform hand-crafted features for FAR. In [24], face detection and attribute recognition CNNs are carefully designed, and the output of the face detection network is fed into the attribute network. An alternative to purpose designing CNNs for FAR is to fine-tune networks intended for object recognition [56, 57]. From a representation learning perspective, the features supporting different attribute detections may be shared, leading some studies to investigate multi-task learning facial attributes [55, 30]. Since different facial attributes have different prevalence, the multi-label/multi-task learning suffers from label-imbalance, which [30] addresses using a mixed objective optimization network (MOON).

Face Recognition using Facial Attributes. Detected facial attributes can be applied directly to authentication. Facial attributes have been applied to enhance face verification, primarily in the case of cross-modal matching, by filtering [19, 54] (requiring potential FRF matches to have the correct gender, for example), model switching [18], or aggregation with conventional features [27, 17]. [21] defines 65 facial attributes and proposes binary attribute classifiers to predict their presence or absence. The vector of attribute classifier scores can be used for face recognition. There has been little work on attribute-enhanced face recognition in the context of deep learning. One of the few exploits CNN-based attribute features for authentication on mobile devices [31]. Local facial patches are fed into carefully designed CNNs to predict different attributes. After CNN training, SVMs are trained for attribute recognition, and the vector of SVM scores provide the new feature for face verification.

Fusion Methods. Existing fusion approaches can be classified into feature-level (early fusion) and score-level (late fusion). Score-level fusion is to fuse the similarity scores after computation based on each view either by simple averaging [37] or stacking another classifier [48, 37]. Feature-level fusion can be achieved by either simple feature aggregation or subspace learning. For aggregation approaches, fusion is usually performed by simply element wise averaging or product (the dimension of features have to be the same) or concatenation [28]. For subspace learning approaches, the features are first concatenated, then the concatenated feature is projected to a subspace, in which the features should better complement each other. These subspace approaches can be unsupervised or supervised. Unsupervised fusion does not use the identity (label) information to learn the subspace, such as Canonical Correlational Analysis (CCA) [35] and Bilinear Models (BLM) [45]. In comparison, supervised fusion uses the identity information such as Linear Discriminant Analysis (LDA) [3] and Locality Preserving Projections (LPP) [9].

Neural Tensor Methods. Learning tensor-based computations within neural networks has been studied for full [39] and decomposed [16, 52, 51] tensors. However, aside from differing applications and objectives, the key difference is that we establish a novel equivalence between a rich Tucker [46] decomposed low-rank fusion tensor, and a gated two-stream neural network. This allows us achieve expressive fusion, while maintaining tractable computation and a small number of parameters; and crucially permits easy optimisation of the fusion tensor through standard toolboxes.

Motivation. Facial attribute features (FAF) and face recognition features (FRF) are complementary. However in practice, we find that existing fusion methods often cannot effectively combine these asymmetric features so as to improve performance. This motivates us to design a more powerful fusion method, as detailed in Section 3. Based on our neural tensor fusion method, in Section 5 we systematically explore the fusion of FAF and FRF in various face recognition environments, showing that FAF can greatly enhance recognition performance.

3. Fusing attribute and recognition features

In this section we present our strategy for fusing FAF and FRF. Our goal is to input FAF and FRF and output the fused discriminative feature. The proposed fusion method we present here performs significantly better than the existing ones introduced in Section 2. In this section, we detail our tensor-based fusion strategy.

3.1. Modelling

Single Feature. We start from a standard multi-class classification problem setting: assume we have M instances, and for each we extract a D-dimensional feature vector (the FRF) as $\{\mathbf{x}^{(i)}\}_{i=1}^{M}$. The label space contains C unique classes (person identities), so each instance is associated with a corresponding C-dimensional one-hot encoding label vector $\{\mathbf{y}^{(i)}\}_{i=1}^{M}$. Assuming a linear model $\mathbf{W}$ the prediction $\hat{\mathbf{y}}_{(i)}$ is produced by the dot-product of input $\mathbf{x}^{(i)}$ and the model $\mathbf{W}$,

$$\hat{\mathbf{y}}^{(i)} = {\mathbf{x}^{(i)}}^{\mathrm{T}}\mathbf{W}. \tag{1}$$

Multiple Feature. Suppose that apart from the D-dimensional FRF vector, we can also obtain an instance-wise B-dimensional facial attribute feature $\mathbf{z}^{(i)}$. Then the input for the ith instance is a pair: $\{\mathbf{x}^{(i)}, \mathbf{z}^{(i)}\}$. A simple approach is to redefine $\mathbf{x}^{(i)} := [\mathbf{x}^{(i)}, \mathbf{z}^{(i)}]$, and directly apply Eq. (1), thus modelling weights for both FRF and FAF features. Here we propose instead a non-linear fusion method via the following formulation

$$\hat{\mathbf{y}}^{(i)} = \mathcal{W} \times_1 \mathbf{x}^{(i)} \times_3 \mathbf{z}^{(i)} \tag{2}$$

where $\mathcal{W}$ is the fusion model parameters in the form of a third-order tensor of size $D \times C \times B$. Notation $\times$ is the tensor dot product (also known as tensor contraction) and the left-subscript of $\mathbf{x}$ and $\mathbf{z}$ indicates at which axis the tensor dot product operates. With Eq. (2), the optimisation problem is formulated as:

$$\min_{\mathcal{W}} \frac{1}{M}\sum_{i=1}^{M} \ell\Big(\mathcal{W} \times_1 \mathbf{x}^{(i)} \times_3 \mathbf{z}^{(i)}, \mathbf{y}^{(i)}\Big) \tag{3}$$

where $\ell(\cdot,\cdot)$ is a loss function. This trains tensor $\mathcal{W}$ to fuse FRF and FAF features so that identity is correctly predicted.

3.2. Optimisation

The proposed tensor $\mathcal{W}$ provides a rich fusion model. However, compared with $\mathbf{W}$, $\mathcal{W}$ is B times larger ($D \times C$ vs $D \times C \times B$) because of the introduction of B-dimensional attribute vector. It is also almost B times larger than training a matrix $\mathbf{W}$ on the concatenation $[\mathbf{x}^{(i)}, \mathbf{z}^{(i)}]$. It is therefore problematic to directly optimise Eq. (3) because the large number of parameters of $\mathcal{W}$ makes training slow and leads to overfitting. To address this we propose a tensor decomposition technique and a neural network architecture to solve an equivalent optimisation problem in the following two subsections.

3.2.1 Tucker Decomposition for Feature Fusion

To reduce the number of parameters of $\mathcal{W}$, we place a structural constraint on $\mathcal{W}$. Motivated by the famous Tucker decomposition [46] for tensors, we assume that $\mathcal{W}$ is synthesised from

$$\mathcal{W} = \mathcal{S} \times_1 \mathbf{U}^{(D)} \times_2 \mathbf{U}^{(C)} \times_3 \mathbf{U}^{(B)}. \tag{4}$$

Here $\mathcal{S}$ is a third order tensor of size $K_D \times K_C \times K_B$, $\mathbf{U}^{(D)}$ is a matrix of size $K_D \times D$, $\mathbf{U}^{(C)}$ is a matrix of size

$K_C \times C$, and $\mathbf{U}^{(B)}$ is a matrix of size $K_B \times B$. By restricting $K_D \ll D$, $K_C \ll C$, and $K_B \ll B$, we can effectively reduce the number of parameters from $(D \times C \times B)$ to $(K_D \times K_C \times K_B + K_D \times D + K_C \times C + K_B \times B)$ if we learn $\{\mathcal{S}, \mathbf{U}^{(D)}, \mathbf{U}^{(C)}, \mathbf{U}^{(B)}\}$ instead of $\mathcal{W}$.

When $\mathcal{W}$ is needed for making the predictions, we can always synthesise it from those four small factors. In the context of tensor decomposition, (K_D, K_C, K_B) is usually called the tensor's rank, as an analogous concept to the rank of a matrix in matrix decomposition.

Note that, despite of the existence of other tensor decomposition choices, Tucker decomposition offers a greater flexibility in terms of modelling because we have three hyper-parameters K_D, K_C, K_B corresponding to the axes of the tensor. In contrast, the other famous decomposition, CP [10] has one hyper-parameter K for *all* axes of tensor.

By substituting Eq. (4) into Eq. (2), we have

$$\begin{aligned}\hat{\mathbf{y}}^{(i)} &= \mathcal{W} \times_1 \mathbf{x}^{(i)} \times_3 \mathbf{z}^{(i)} \\ &= \mathcal{S} \times_1 \mathbf{U}^{(D)} \times_2 \mathbf{U}^{(C)} \times_3 \mathbf{U}^{(B)} \times_1 \mathbf{x}^{(i)} \times_3 \mathbf{z}^{(i)} \end{aligned} \quad (5)$$

Through some re-arrangement, Eq. (5) can be simplified as

$$\hat{\mathbf{y}}^{(i)} = \mathcal{S} \times_1 (\mathbf{U}^{(D)}\mathbf{x}^{(i)}) \times_2 \mathbf{U}^{(C)} \times_3 (\mathbf{U}^{(B)}\mathbf{z}^{(i)}) \quad (6)$$

Furthermore, we can rewrite Eq. (6) as,

$$\hat{\mathbf{y}}^{(i)} = \underbrace{((\mathbf{U}^{(D)}\mathbf{x}^{(i)}) \otimes (\mathbf{U}^{(B)}\mathbf{z}^{(i)}))\mathcal{S}_{(2)}^T}_{\textbf{fused feature}} \mathbf{U}^{(C)} \quad (7)$$

where $\otimes$ is Kronecker product. Since $\mathbf{U}^{(D)}\mathbf{x}^{(i)}$ and $\mathbf{U}^{(B)}\mathbf{B}^{(i)}$ result in K_D and K_B dimensional vectors respectively, $(\mathbf{U}^{(D)}\mathbf{x}^{(i)}) \otimes (\mathbf{U}^{(B)}\mathbf{z}^{(i)})$ produces a $K_D K_B$ vector. $\mathcal{S}_{(2)}$ is the mode-2 unfolding of $\mathcal{S}$ which is a $K_C \times K_D K_B$ matrix, and its transpose $\mathcal{S}_{(2)}^T$ is a matrix of size $K_D K_B \times K_C$.

The Fused Feature. From Eq. (7), the explicit fused representation of face recognition $(\mathbf{x}^{(i)})$ and facial attribute $(\mathbf{z}^{(i)})$ features can be achieved. The fused feature $((\mathbf{U}^{(D)}\mathbf{x}^{(i)}) \otimes (\mathbf{U}^{(B)}\mathbf{z}^{(i)}))\mathcal{S}_{(2)}^T$, is a vector of the dimensionality K_C. And matrix $\mathbf{U}^{(C)}$ has the role of "classifier" given this fused feature. Given $\{\mathbf{x}^{(i)}, \mathbf{z}^{(i)}, \mathbf{y}^{(i)}\}$, the matrices $\{\mathbf{U}^{(D)}, \mathbf{U}^{(B)}, \mathbf{U}^{(C)}\}$ and tensor $\mathcal{S}$ are computed (learned) during model optimisation (training). During testing, the prediction $\hat{\mathbf{y}}^{(i)}$ is achieved with the learned $\{\mathbf{U}^{(D)}, \mathbf{U}^{(B)}, \mathbf{U}^{(C)}, \mathcal{S}\}$ and two test features $\{\mathbf{x}^{(i)}, \mathbf{z}^{(i)}\}$ following Eq. (7).

3.2.2 Gated Two-stream Neural Network (GTNN)

A key advantage of reformulating Eq. (5) into Eq. (7) is that we can now find a neural network architecture that does exactly the computation of Eq. (7), which would not be obvious if we stopped at Eq. (5). Before presenting this neural network, we need to introduce a new deterministic layer (i.e. without any learnable parameters).

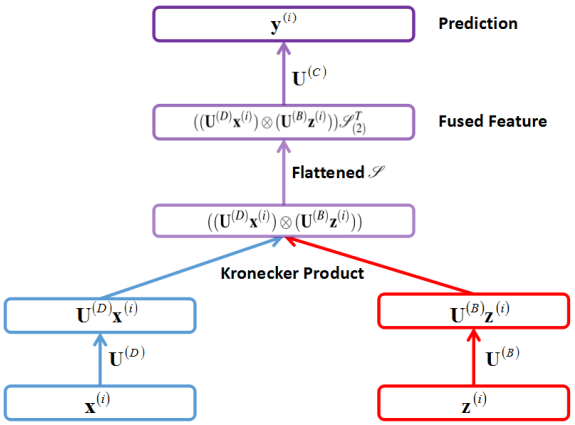

Figure 2: Gated two-stream neural network to implement low-rank tensor-based fusion. The architecture computes Eq. (7), with the Tucker decomposition in Eq. (4). The network is identity-supervised at train time, and feature in the fusion layer used as representation for verification.

Kronecker Product Layer takes two arbitrary-length input vectors $\{\mathbf{u}, \mathbf{v}\}$ where $\mathbf{u} = [u_1, u_2, \cdots, u_P]$ and $\mathbf{v} = [v_1, v_2, \cdots, v_Q]$, then outputs a vector of length PQ as $[u_1 v_1, u_1 v_2, \cdots, u_1 v_Q, u_2 v_1, \cdots, u_P v_Q]$.

Using the introduced Kronecker layer, Fig. 2 shows the neural network that computes Eq. (7). That is, the neural network that performs recognition using tensor-based fusion of two features (such as FAF and FRF), based on the low-rank assumption in Eq. (4). We denote this architecture as a Gated Two-stream Neural Network (GTNN), because it takes two streams of inputs, and it performs gating [36] (multiplicative) operations on them.

The GTNN is trained in a supervised fashion to predict identity. In this work, we use a multitask loss: softmax loss and center loss [47] for joint training. The fused feature in the viewpoint of GTNN is the output of penultimate layer, which is of dimensionality K_c.

So far, the advantage of using GTNN is obvious. Direct use of Eq. (5) or Eq. (7) requires manual derivation and implementation of an optimiser which is non-trivial even for decomposed matrices (2d-tensors) [20]. In contrast, GTNN is easily implemented with modern deep learning packages where auto-differentiation and gradient-based optimisation is handled robustly and automatically.

3.3. Discussion

Compared with the fusion methods introduced in Section 2, we summarise the advantages of our tensor-based fusion method as follows:

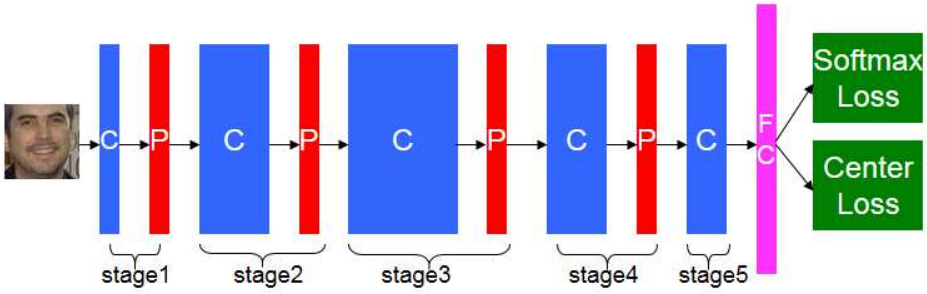

Figure 3: LeanFace. 'C' is a group of convolutional layers. Stage 1: 64 @ 5×5 (64 feature maps are sliced to two groups of 32 ones, which are fed into maxout function.) ; Stage 2: 64 @ 3×3, 64 @ 3×3, 128 @ 3×3, 128 @ 3×3; Stage 3: 196 @ 3×3, 196 @ 3×3, 256 @ 3×3, 256 @ 3×3, 320 @ 3×3, 320 @ 3×3; Stage 4: 512 @ 3×3, 512 @ 3×3, 512 @ 3×3, 512 @ 3×3; Stage 5: 640 @ 5×5, 640@ 5×5. 'P' stands for 2×2 max pooling. The strides for the convolutional and pooling layers are 1 and 2, respectively. 'FC' is a fully-connected layer of 256D.

High Order Non-Linearity. Unlike linear methods based on averaging, concatenation, linear subspace learning [8, 27], or LDA [3], our fusion method is non-linear, which is more powerful to model complex problems. Furthermore, comparing with other first-order non-linear methods based on element-wise combinations only [28], our method is higher order: it accounts for all interactions between each pair of feature channels in both views. Thanks to the low-rank modelling, our method achieves such powerful non-linear fusion with few parameters and thus it is robust to overfitting.

Scalability. Big datasets are required for state-of-the-art face representation learning. Because we establish the equivalence between tensor factorisation and gated neural network architecture, our method is scalable to big-data through efficient mini-batch SGD-based learning. In contrast, kernel-based non-linear methods, such as Kernel LDA [34] and multi-kernel SVM [17], are restricted to small data due to their $O(N^2)$ computation cost. At runtime, our method only requires a simple feed-forward pass and hence it is also favourable compared to kernel methods.

Supervised method. GTNN is flexibly supervised by any desired neural network loss function. For example, the fusion method can be trained with losses known to be effective for face representation learning: identity-supervised softmax, and centre-loss [47]. Alternative methods are either unsupervised [8, 27], constrained in the types of supervision they can exploit [3, 17], or only stack scores rather than improving a learned representation [48, 37]. Therefore, they are relatively ineffective at learning how to combine the two-source information in a task-specific way.

Extensibility. Our GTNN naturally can be extended to deeper architectures. For example, the pre-extracted features, i.e., $\mathbf{x}$ and $\mathbf{z}$ in Fig. 2, can be replaced by two full-sized CNNs without any modification. Therefore, potentially, our methods can be integrated into an end-to-end framework.

4. Integration with CNNs: architecture

In this section, we introduce the CNN architectures used for face recognition (LeanFace) designed by ourselves and facial attribute recognition (AttNet) introduced by [50, 30].

LeanFace. Unlike general object recognition, face recognition has to capture very subtle difference between people. Motivated by the fine-grain object recognition in [4], we also use a large number of convolutional layers at early stage to capture the subtle low level and mid-level information. Our activation function is maxout, which shows better performance than its competitors [50]. Joint supervision of softmax loss and center loss [47] is used for training. The architecture is summarised in Fig. 3.

AttNet. To detect facial attributes, our AttNet uses the architecture of Lighten CNN [50] to represent a face. Specifically, AttNet consists of 5 conv-activation-pooling units followed by a 256D fully connected layer. The number of convolutional kernels is explained in [50]. The activation function is Max-Feature-Map [50] which is a variant of maxout. We use the loss function MOON [30], which is a multi-task loss for (1) attribute classification and (2) domain adaptive data balance. In [24], an ontology of 40 facial attributes are defined. We remove attributes which do not characterise a specific person, e.g., 'wear glasses' and 'smiling', leaving 17 attributes in total.

Once each network is trained, the features extracted from the penultimate fully-connected layers of LeanFace (256D) and AttNet (256D) are extracted as $\mathbf{x}$ and $\mathbf{z}$, and input to GTNN for fusion and then face recognition.

5. Experiments

We first introduce the implementation details of our GTNN method. In Section 5.1, we conduct experiments on MultiPIE [7] to show that facial attributes by means of our GTNN method can play an important role on improv-

Table 1: Network training details

	Image size	Batch size	LR [1]	DF [2]	Epoch	Train time
LeanFace	128 x 128	256	0.001	0.1	54	91h
AttNet			0.05	0.8	99	3h

[1] Learning rate (LR)
[2] Learning rate drop factor (DF).

ing face recognition performance in the presence of pose, illumination and expression, respectively. Then, we compare our GTNN method with other fusion methods on CASIA NIR-VIS 2.0 database [22] in Section 5.2 and LFW database [12] in Section 5.3, respectively.

Implementation Details. In this study, three networks (LeanFace, AttNet and GTNN) are discussed. LeanFace and AttNet are implemented using MXNet [6] and GTNN uses TensorFlow [1]. We use around 6M training face thumbnails covering 62K different identities to train LeanFace, which has no overlapping with all the test databases. AttNet is trained using CelebA [24] database. The input of GTNN is two 256D features from bottleneck layers (i.e., fully connected layers before prediction layers) of LeanFace and AttNet. The setting of main parameters are shown in Table 1. Note that the learning rates drop when the loss stops decreasing. Specifically, the learning rates change 4 and 2 times for LeanFace and AttNet respectively. During test, LeanFace and AttNet take around 2.9ms and 3.2ms to extract feature from one input image and GTNN takes around 2.1ms to fuse one pair of LeanFace and AttNet feature using a GTX 1080 Graphics Card.

5.1. Multi-PIE Database

Multi-PIE database [7] contains more than 750,000 images of 337 people recorded in 4 sessions under diverse pose, illumination and expression variations. It is an ideal testbed to investigate if facial attribute features (FAF) complement face recognition features (FRF) including traditional hand-crafted (LBP) and deeply learned features (LeanFace) to improve the face recognition performance – particularly across extreme pose variation.

Settings. We conduct three experiments to investigate pose-, illumination- and expression-invariant face recognition. *Pose:* Uses images across 4 sessions with pose variations only (i.e., neutral lighting and expression). It covers pose with yaw ranging from left 90° to right 90°. In comparison, most of the existing works only evaluate performance on poses with yaw range (-45°, +45°). *Illumination:* Uses images with 20 different illumination conditions (i.e., frontal pose and neutral expression). *Expression:* Uses images with 7 different expression variations (i.e., frontal pose and neutral illumination). The training sets of all settings consist of the images from the first 200 subjects and the remaining 137 subjects for testing. Following [59, 14], in the test set, frontal images with neural illumination and expression from the earliest session work as gallery, and the others are probes.

Pose. Table 2 shows the pose-robust face recognition (PRFR) performance. Clearly, the fusion of FRF and FAF, namely GTNN (LBP, AttNet) and GTNN (LeanFace, AttNet), works much better than using FRF only, showing the complementary power of facial features to face recognition features. Not surprisingly, the performance of both LBP and LeanFace features drop greatly under extreme poses, as pose variation is a major factor challenging face recognition performance. In contrast, with GTNN-based fusion, FAF can be used to improve *both* classic (LBP) and deep (LeanFace) FRF features effectively under this circumstance, for example, LBP (1.3%) vs GTNN (LBP, AttNet) (16.3%), LeanFace (72.0%) vs GTNN (LeanFace, AttNet) (78.3%) under yaw angel $-90°$. It is noteworthy that despite their highly asymmetric strength, GTNN is able to effectively fuse FAF and FRF. This is elaborately studied in more detail in Sections 5.2-5.3.

Compared with state-of-the-art methods [14, 59, 11, 58, 15] in terms of (-45°, +45°), LeanFace achieves better performance due to its big training data and the strong generalisation capacity of deep learning. In Table 2, 2D methods [14, 59, 15] trained models using the MultiPIE images, therefore, they are difficult to generalise to images under poses which do not appear in MultiPIE database. 3D methods [11, 58] highly depend on accurate 2D landmarks for 3D-2D modelling fitting. However, it is hard to accurately detect such landmarks under larger poses, limiting the applications of 3D methods.

Illumination and expression. Illumination- and expression-robust face recognition (IRFR and ERFR) are also challenging research topics. LBP is the most widely used handcrafted features for IRFR [2] and ERFR [33]. To investigate the helpfulness of facial attributes, experiments of IRFR and ERFR are conducted using LBP and LeanFace features. In Table 3, GTNN (LBP, AttNet) significantly outperforms LBP, 80.3% vs 57.5% (IRFR), 77.5% vs 71.7% (ERFR), showing the great value of combining facial attributes with hand-crafted features. Attributes such as the shape of eyebrows are illumination invariant and others, e.g., gender, are expression invariant. In contrast, LeanFace feature is already very discriminative, saturating the performance on the test set. So there is little room for fusion of AttrNet to provide benefit.

5.2. CASIA NIR-VIS 2.0 Database

The CASIA NIR-VIS 2.0 face database [22] is the largest public face database across near-infrared (NIR) images and visible RGB (VIS) images. It is a typical cross-modality or heterogeneous face recognition problem because the gallery and probe images are from two different spectra. The

Table 2: Face recognition rate (%) on different poses on Multi-PIE

Method	-90°	-75°	-60°	(-45°, 45°)	+60°	+75°	+90°
SPAE [14]		-		91.4		-	
RL [59]		-		98.3		-	
MvDN [15]		-		99.3		-	
U3DMM [11]		-		97.8		-	
E3DMM [58]		-		98.6		-	
LBP	1.3	2.3	1.7	43.0	0.7	0.3	0.7
LeanFace	72.0	94.3	99.0	100	98.3	89.0	61.0
AttNet	6.0	9.7	11.7	56.1	11.3	8.0	5.0
GTNN (LBP, AttNet)	**16.3**	**14.3**	**15.0**	**69.3**	**6.7**	**4.3**	**3.3**
GTNN (LeanFace, AttNet)	**78.3**	**97.3**	**99.7**	**100**	**98.0**	**94.3**	**68.0**

Table 3: Average face recognition rate (%) on different illuminations and expressions on Multi-PIE

Method	Illumination	Expression
LBP	57.5	71.7
LeanFace	100	99.8
AttNet	53.8	48.8
GTNN (LBP, AttNet)	**80.3**	**77.5**
GTNN (LeanFace, AttNet)	**100**	**100**

gallery and probe images are VIS and NIR images respectively. It simulates the scenario of face recognition in a dark environment, where only NIR images are available for probing. This database consists of 17,580 images of 725 subjects which exhibit intra-personal variations such as pose and expression. Similar to most face databases, CASIA NIR-VIS 2.0 includes two views: view 1 for training and view 2 including 10 folds for performance evaluation. Following the standard evaluation protocol, the rank 1 identification rate of 10 folds is reported.

Comparison with State-of-the-art. As shown in Table 4, LeanFace and Light CNN [49] already achieve very impressive performance due to their big training data and effective deep learning architectures. It is noteworthy that the gallery and probe are VIS and NIR images respectively, while LeanFace and Light CNN are trained using *only* VIS images. Their efficacy here shows that CNNs trained using big data learn a sufficiently robust face representation which bridges the gap between VIS and NIR. CNN architectures + big VIS training images greatly outperforms hand-crafted features + explicit cross modality learning models [25, 13, 53], suggesting that explicit cross-modal learning might be unnecessary for VIS-NIR. Comparing with the CNNs, LeanFace works better than Light CNN because it uses (1) larger training data (6M vs 5M) (2) better loss functions (softmax + centerloss vs softmax) and (3) deeper architectures. GTNN (LeanFace and AttNet) works better than LeanFace, 99.94% vs 97.27%, meaning that facial attributes are complementary with the LeanFace feature in NIR-VIS cross-modality face recognition.

Table 4: Comparison with State-of-the-art on CASIA NIR-VIS 2.0 face database

Method	Acc.(%)
C-CBFD+LDA [25]	81.8
Dictionary Learning [13]	78.46
Gabor+RBM [53]	86.16
Light CNN [49]	91.88
LeanFace	97.27
AttNet	2.38
GTNN (LeanFace, AttNet)	**99.94**

Comparison with other fusion methods. In the previous experiment, GTNN successfully fused LeanFace and AttNet, despite their extreme asymmetry in individual strength. In this experiment, we verify that this is a non-trivial achievement, by comparing with other popular fusion methods, shown in Table 5.

Simple concatenation and average fusion achieve the same accuracy 97.27% as using LeanFace feature only. Clearly, the stronger feature (i.e., LeanFace) dominates the fused feature. Another three unsupervised fusion methods: score fusion, CCA [35] and BLM [45] achieve the accuracy between using LeanFace only and AttNet only. This outcome of a weaker feature making the fused feature worse is common when fusing very asymmetric features. The three supervised fusion methods achieve higher accuracy than LeanFace, showing the importance of label information for fusion. Supervised fusion methods can also be viewed as metric learning, which has been proven effective for various face recognition scenarios [9, 3, 44]. Nevertheless, the proposed GTNN (99.94%) works better than LDA (98.33%) and LPP (98.58%) due to its stronger non-linear modelling capacity. Finally, we reiterate that GTNN has the potential to work with CNNs for end-to-end training, while LDA and

LPP cannot.

Table 5: Comparison with fusion methods on NIR-VIS 2.0

Method		Acc. (%)
Raw Feature	LeanFace	97.27
	AttNet	2.38
Unsupervised Fusion	Concatenation	97.27
	Average	97.27
	Score	64.24
	CCA [35]	93.80
	BLM [45]	90.57
Supervised Fusion	LDA [3]	98.33
	LPP [9]	98.58
	GTNN	**99.94**

5.3. LFW Database

Face recognition in uncontrolled environments (FRUE) is widely studied in recent years. LFW [12] is the most widely used FRUE benchmark which contains 13,233 images of 5,749 subjects. For evaluation, LFW is divided into 10 predefined splits for cross validation. We follow the standard 'Unrestricted, Labelled Outside Data Results' protocol [12] for testing. To train the fusion methods, we use 0.1M images of 1.5K subjects (non-overlapping with LFW subjects) from our training data for LeanFace.

Comparison with State-of-the-art. LeanFace achieves very promising face recognition rate 99.57%, benefiting from its effective architecture (Fig. 3) and larger training data. Although LeanFace almost saturates the LFW database, the fusion of attribute feature further reduces the error rate by 19%. Our full method (GTNN Fusion) achieves state-of-the-art face recognition rate on LFW. Compared with the two best models (FaceNet [32], DeepID2+ [42]), we do not use network ensemble, while DeepID2+ makes use of an ensemble of 25 CNNs. Facenet uses more than 100M images for training, while we only use 6M images. In addition, Facenet uses triplet loss for metric learning, which is very difficult to sample hard training image triplet, while we use center-loss [47] which does not need to do such sampling. Note that the LFW official website publishes some other promising results that are mostly from industry. However, their methodological details are not published, therefore, we do not compare with them.

Comparison with other fusion methods. In this experiment, the performance of LeanFace only is comparable to AttNet only (99.57% v.s. 79.07%), unlike their performance in the NIR-VIS experiment (97.27% v.s. 2.38%). However, LeanFace has already achieved very high recognition rate, almost saturating the benchmark, making it challenging to further improve the performance. In Table 7, all alternative methods fail to improve the fused performance beyond that of the dominant LeanFace feature. Even the supervised methods LDA and LPP fail to improve as LeanFace is already strong. Score-fusion and CCA [35] make the performance worse compared to LeanFace. Unlike all the alternatives, GTNN further improves the performance because of its powerful non-linear modelling capacity.

Table 6: Comparisons with the state-of-the-art on LFW

Method	Err.Rate(%)	Acc. (%)
DeepFace [44]	2.65	97.35
VGGFace [29]	1.05	98.95
Center loss [47]	0.72	99.28
DeepID2+ [42]	0.53	99.47
FaceNet [32]	0.37	99.63
LeanFace	0.43	99.57
AttNet	20.93	79.07
GTNN (LeanFace, AttNet)	**0.35**	**99.65**

Table 7: Comparison with fusion methods on LFW

Method		Err.Rate(%)	Acc. (%)
Raw Feature	LeanFace	0.43	99.57
	AttNet	20.93	79.07
Unsupervised Fusion	Concatenation	0.43	99.57
	Average	0.43	99.57
	Score	4.31	95.69
	CCA [35]	0.77	99.23
	BLM [45]	0.43	99.57
Supervised Fusion	LDA [3]	0.43	99.57
	LPP [9]	0.43	99.57
	GTNN	**0.35**	**99.65**

6. Conclusion

We considered the problem of enhancing face recognition by incorporating predicted attributes. This provides additional robustness to complicated intra-personal variations in face recognition. We presented a powerful non-linear tensor-based fusion method that can synergistically combine attribute-derived features with both hand-crafted and deep conventional features. Our method is both easy to implement and efficient to train due to our establishment of a correspondence to a specific neural network architecture.

Acknowledgement We acknowledge the support of the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/K014277/1, the MOD University Defence Research Collaboration (UDRC) in Signal Processing, the European Union's Horizon 2020 research and innovation program under grant agreement No 640891, National Natural Science Foundation of China (Grant No.61402389) and the Fundamental Research Funds for the Central Universities (No. 20720160073).

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. J. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Józefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. G. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. A. Tucker, V. Vanhoucke, V. Vasudevan, F. B. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *CoRR*, abs/1603.04467, 2016. 6

[2] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE TPAMI*, 2006. 6

[3] P. N. Belhumeur, J. Hespanha, o P, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *TPAMI*, 1996. 3, 5, 7, 8

[4] X. Cao. A practical theory for designing very deep convolutional neural networks. *Technical Report*, 2015. 5

[5] D. Chen, X. Cao, F. Wen, and J. Sun. Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. In *CVPR*, 2013. 2

[6] T. Chen. Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. In *Workshop on Machine Learning Systems*, 2015. 6

[7] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image and Vision Computing*, 2010. 5, 6

[8] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16(12):2639–2664, 2004. 5

[9] X. He, S. Yan, Y. Hu, P. Niyogi, and H. J. Zhang. Face recognition using laplacianfaces. *IEEE TPAMI*, 2005. 3, 7, 8

[10] F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics*, 6(1):164–189, 1927. 4

[11] G. Hu, F. Yan, C.-H. Chan, W. Deng, W. Christmas, J. Kittler, and N. M. Robertson. Face recognition using a unified 3d morphable model. In *ECCV*, 2016. 6, 7

[12] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, 2007. 6, 8

[13] F. Juefei-Xu, D. K. Pal, and M. Savvides. Nir-vis heterogeneous face recognition via cross-spectral joint dictionary learning and reconstruction. In *CVPRW*, 2015. 7

[14] M. Kan, S. Shan, H. Chang, and X. Chen. Stacked progressive auto-encoders (spae) for face recognition across poses. In *CVPR*, 2013. 6, 7

[15] M. Kan, S. Shan, and X. Chen. Multi-view deep network for cross-view classification. In *CVPR*, 2016. 6, 7

[16] R. Kiros, R. Salakhutdinov, and R. Zemel. Multimodal neural language models. In *ICML*, 2014. 3

[17] B. Klare, S. Bucak, A. Jain, and T. Akgul. Towards automated caricature recognition. In *ICB*, 2012. 2, 5

[18] B. Klare, M. Burge, J. Klontz, R. Vorder Bruegge, and A. Jain. Face recognition performance: Role of demographic information. *TIFS*, 2012. 2

[19] B. Klare, Z. Li, and A. Jain. Matching forensic sketches to mug shot photos. *TPAMI*, 2011. 2

[20] A. Kumar and H. Daumé III. Learning task grouping and overlap in multi-task learning. In *ICML*, 2012. 4

[21] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, 2009. 2

[22] S. Li, D. Yi, Z. Lei, and S. Liao. The casia nir-vis 2.0 face database. In *CVPRW*, 2013. 6

[23] C. Liu and H. Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Transactions on Image Processing*, 2002. 2

[24] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 2, 5, 6

[25] J. Lu, V. E. Liong, X. Zhou, and J. Zhou. Learning compact binary face descriptor for face recognition. *TPAMI*, 2015. 7

[26] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *TPAMI*, 2002. 2

[27] S. Ouyang, T. M. Hospedales, Y.-Z. Song, and X. Li. Cross-modal face matching: Beyond viewed sketches. In *ACCV*, 2014. 2, 5

[28] E. Park, X. Han, T. L. Berg, and A. C. Berg. Combining multiple sources of knowledge in deep cnns for action recognition. In *WACV*, 2016. 3, 5

[29] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *BMVC*, 2015. 2, 8

[30] E. M. Rudd, M. Günther, and T. E. Boult. MOON: A mixed objective optimization network for the recognition of facial attributes. In *ECCV*, 2016. 2, 5

[31] P. Samangouei and R. Chellappa. Convolutional neural networks for attribute-based active authentication on mobile devices. *arXiv:1604.08865*, 2016. 2

[32] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *arXiv preprint arXiv:1503.03832*, 2015. 1, 2, 8

[33] C. Shan, S. Gong, and P. W. McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing*, 2009. 6

[34] A. Sharma, A. Kumar, H. Daume, and D. W. Jacobs. Generalized multiview analysis: A discriminative latent space. In *CVPR*. IEEE, 2012. 1, 5

[35] J. Shawe-Taylor and N. Cristianini. Kernel methods for pattern analysis. *Journal of the American Statistical Association*, 2004. 3, 7, 8

[36] O. Sigaud, C. Masson, D. Filliat, and F. Stulp. Gated networks: an inventory. *arXiv*, 2015. 4

[37] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*. 2014. 3, 5

[38] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[39] R. Socher, D. Chen, C. D. Manning, and A. Y. Ng. Reasoning with neural tensor networks for knowledge base completion. In *NIPS*, 2013. 3

[40] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *NIPS*, 2014. 1, 2

[41] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *CVPR*, 2014. 1, 2

[42] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. *arXiv preprint arXiv:1412.1265*, 2014. 1, 2, 8

[43] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*, 2014. 2

[44] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, 2014. 2, 7, 8

[45] J. B. Tenenbaum and W. T. Freeman. Separating style and content with bilinear models. *Neural Computation*, 2000. 3, 7, 8

[46] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 1966. 3

[47] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, 2016. 2, 4, 5, 8

[48] D. H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241 – 259, 1992. 3, 5

[49] X. Wu, R. He, Z. Sun, and T. Tan. A light cnn for deep face representation with noisy labels. *arXiv preprint arXiv:1511.02683*, 2015. 7

[50] Z. S. Xiang Wu, Ran He. A lightened cnn for deep face representation. *arXiv:1511.02683*, 2015. 5

[51] Y. Yang and T. M. Hospedales. Deep multi-task representation learning: A tensor factorisation approach. In *ICLR*, 2017. 3

[52] Y. Yang and T. M. Hospedales. Unifying multi-domain multi-task learning: Tensor and neural network perspectives. In G. Csurka, editor, *Domain Adaptation in Computer Vision Applications*. Springer, 2017. 3

[53] D. Yi, Z. Lei, and S. Z. Li. Shared representation learning for heterogenous face recognition. In *FG*, 2015. 7

[54] H. Zhang, J. R. Beveridge, B. A. Draper, and P. J. Phillips. On the effectiveness of soft biometrics for increasing face verification rates. *CVIU*, 2015. 2

[55] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Facial landmark detection by deep multi-task learning. In *ECCV*, 2014. 2

[56] Y. Zhong, J. Sullivan, and H. Li. Face attribute prediction using off-the-shelf cnn features. In *ICB*, 2016. 2

[57] Y. Zhong, J. Sullivan, and H. Li. Leveraging mid-level deep representations for predicting face attributes in the wild. In *ICIP*, 2016. 2

[58] X. Zhu, Z. Lei, J. Yan, D. Yi, and S. Z. Li. High-fidelity pose and expression normalization for face recognition in the wild. In *CVPR*, 2015. 6, 7

[59] Z. Zhu, P. Luo, X. Wang, and X. Tang. Deep learning identity-preserving face space. In *ICCV*, 2013. 6, 7